

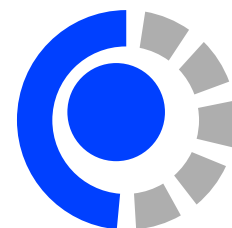
# Trabajo práctico de laboratorio N° 10

## Introducción al shell del sistema GNU/LINUX

FECHA DE FINALIZACIÓN: 10 DE JUNIO



Introducción a la computación  
Departamento de Ingeniería de Computadoras  
Facultad de Informática - Universidad Nacional del Comahue



### Lectura obligatoria:

- Apunte del shell de Linux: <http://pedco.uncoma.edu.ar/mod/resource/view.php?id=207175>
- Apunte introductorio a **BASH**: <http://pedco.uncoma.edu.ar/mod/resource/view.php?id=244968>
- Linux Man Pages Online: <https://linux.die.net/man/>

A continuación, se realizarán una serie de ejercicios para los cuales necesitará acceso a una computadora con el sistema operativo Linux (o una Máquina Virtual corriendo Linux) y las siguientes aplicaciones:

- Intérprete de comandos **BASH** y demás utilidades que se encuentran en la mayoría de las distribuciones de Linux.
- Un editor de texto en la interfaz de línea de comando o gráfica.

NOTA: para un tutorial de conexión remota desde Windows a su usuario de alumno FI (creado por la Facultad de Informática oportunamente), dirijase a 1.3 Anexo I: Conexión desde entorno Windows a usuario de alumno FI

NOTA 2: para un tutorial de conexión remota desde Linux a su usuario de alumno FI (creado por la Facultad de Informática oportunamente), dirijase a 1.4 Anexo II: Conexión desde entorno Linux a usuario de alumno FI

El sistema operativo controla diferentes procesos de una computadora. Uno de ellos es el intérprete de comandos o *shell*. Este es un programa que permite al usuario interactuar con el Sistema Operativo. Permite iniciar (ejecutar) otros programas así como también tiene comandos propios que no necesitan de otros programas, como por ejemplo comandos para movernos en la estructura de directorios. En este práctico, utilizaremos el intérprete **BASH** (*Bourne-Again SHell*), muy popular en el mundo de Linux.

Dicho programa debe ser ejecutado en lo que llamamos Terminal o Consola, que es otro programa que nos permite interactuar mediante el teclado (ingresar caracteres) y ver los resultados de la ejecución de otros programas en la pantalla. Al iniciar una consola o terminal de textos, automáticamente se inicia el *shell BASH*, que es con el que el usuario realmente interactúa (más abajo se explica esto con un ejemplo).

## 1. Sistemas de archivos

El concepto de directorios (comúnmente llamados “carpetas”) y archivos es hoy en día familiar a los usuarios de computadoras, en general el usuario se maneja visualmente con el ratón y el Explorador de Archivos, utilizando estos para acceder a los directorios, abrir archivos, ejecutar programas, etc. Podemos pensar el sistema de archivos como una forma de organizar todo el contenido en el dispositivo de almacenamiento. Los archivos son datos concretos, utilizan espacio físico del dispositivo, mientras que los directorios sirven para organizar “lógicamente” las cosas, igual que una persona tiene estanterías y cajones en un escritorio para organizar sus papeles.

La figura 1 muestra un esquema de como se puede pensar en una estructura de directorios y archivos. A, B, C, D, E, G y J son directorios, F, H, I, K, L, M, N y O son archivos.

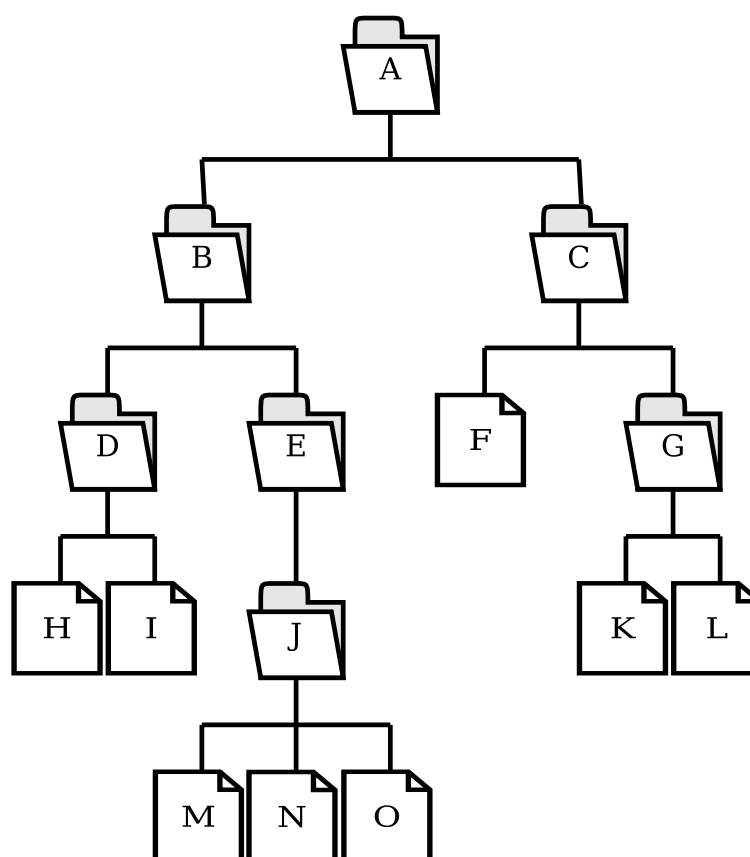


Figura 1: Árbol de directorios.

Cuando se inicia el intérprete de comandos, éste se posiciona en una determinada carpeta, es decir, los comandos que se escriben afectarán directamente al contenido de dicha carpeta.

En general, al iniciar una consola de texto, junto con un intérprete de comandos, se posiciona al usuario en la carpeta personal o “**HOME**” (en Linux en general es “*/home/usuarioX*”). Desde aquí el usuario puede navegar por sus archivos y ejecutar comandos.

Además, existen dos formas de hacer referencia a los directorios y a los archivos en el intérprete, absoluta y relativa. Cuando utilizamos la forma absoluta se escribe todo el camino desde la carpeta inicial (en *Linux* se denota con el símbolo “/”) hasta la ubicación deseada. Ejemplos de uso absoluto: */home/usuario/trabajo1*, */etc*, */tmp*, etc.

El modo relativo sirve para hacer referencia a un directorio que se encuentra “cerca” del directorio de trabajo actual en la jerarquía, como por ejemplo el directorio directamente superior o inferior.

Por ejemplo, las siguientes son direcciones relativas:

- ../ (directorio inmediatamente superior)
- ./trabajo1 (directorio trabajo1 ubicado dentro del directorio actual)
- ../../archivoX (archivo ubicado dos directorios más arriba en la jerarquía)

## 1.1. Entrando en calor con el *shell BASH*

1. Inicie el programa mate terminal: **Aplicaciones** → **Herramientas del sistema** → **Terminal de mate**.

Una vez iniciado, el *shell BASH* en la terminal espera que se escriban órdenes para ejecutar (comandos). Ejecute y analice cuidadosamente la siguiente secuencia de comandos en el intérprete *BASH* (luego de los comandos y parámetros debe presionar *ENTER*).

Puede consultar el manual de cada uno de los comandos utilizados, usando el comando “*man nombre\_del\_comando*”, donde se muestran opciones de uso y se explica la funcionalidad del mismo.

Comandos y parámetros	Descripción
mkdir e	#crear un directorio de nombre “e”
cd e	#cambiar el directorio actual a “e”
mkdir h1	#crear un directorio “h1”
mkdir h2	#crear un directorio “h2”
cd h1	#cambiar el directorio actual a “h1”
nano a.txt	#crear un archivo de texto y guardarlo
pwd	#imprime el directorio actual
cd ..	#cambiar el directorio actual al directorio padre
pwd	#imprime el directorio actual
cd ..	#cambiar el directorio actual al directorio padre
pwd	#imprime el directorio actual
ls	#listado del directorio actual
ls -R	#listado recursivo de los contenidos
mv e/h1/a.txt e/h2/a.txt	#mueve el archivo a.txt a otro directorio
rm e/h2/ab.txt	#eliminar el archivo ab.txt, falla por que no existe
rm e/h2/a.txt	#eliminar el archivo a.txt
rmdir e/h1	#elimina el directorio h1

2. Realice un esquema visual como el de la figura 1, que muestre la estructura de directorios resultante de ejecutar la siguiente secuencia de comandos:

```
mkdir e
cd e
mkdir h1
mkdir h2
```

```
cd h1
nano README
cd ..
cd h2
mkdir h2 h3
cd h2
nano otroArchivo
cd ../../
nano a3.txt
```

3. Utilizando los comandos vistos en el listado, reproduzca la estructura de directorios y archivos de la figura 1, respetando mayúsculas y nombres de los archivos. Esta estructura será utilizada en los subsiguientes incisos.
4. Utilizando la jerarquía de directorios generados en el inciso anterior, posicione el intérprete en el directorio “A”, y realice las siguientes acciones:
  - a) Elimine el archivo “M” sin cambiar de directorio.
  - b) Mueva los archivos “H” e “I” a la carpeta “G”.
  - c) Elimine la carpeta “D”.
  - d) Posicione el intérprete en el directorio “G”.
  - e) Sin cambiar de directorio, listar el contenido del directorio “J” usando directorios relativos.
  - f) Sin cambiar de directorio, listar el contenido del directorio “J” usando directorios absolutos. Para armar una dirección absoluta puede usar el comando “pwd” para conocer como se conforma el camino completo hasta la ubicación actual.
5. Explique las ventajas del uso de direcciones “relativas” a la hora de hacer referencia a otros archivos cercanos en la jerarquía al directorio de trabajo actual. Utilice un ejemplo.
6. Liste el contenido del directorio “J”, con el comando “ls -l *camino\_a\_J*”. Analice la información que se muestra por pantalla, fecha de creación de los archivos, propietario de los archivos y tamaño de los archivos. Modifique alguno de los archivos en el editor de texto y vuelva a analizar la salida del comando “ls -l”.
7. Con el comando “ls -l” encuentre un archivo dentro del directorio “/etc” cuyo tamaño sea mayor a 4KiB, y liste el contenido de dicho archivo utilizando los comandos “cat”, “more” y “less”.

Teclas útiles:

- cat: no tiene.
  - more: “q” para salir, “barra espaciadora” avanzar página.
  - less: “q” para salir, “barra espaciadora” avanzar página, y “j”, “k” para subir y bajar.
8. Retorne a su directorio **HOME**. Para esto ejecute: (**Atención:** el signo \$ a partir de ahora representa el *prompt* del *shell* **BASH**, no debe escribirlo al ingresar cada comando)

```
$ cd          # este es el comando para retornar a su HOME
$ pwd        # pwd le indica cuál es su directorio de trabajo actual
```

## 1.2. Archivos: nombre, tamaño, tipo, contenido.

1. Asegúrese de que su directorio de trabajo actual es su directorio **HOME** (comando: *pwd*).
2. Crear un directorio llamado “misc” (sin las comillas).
3. Ingresar al directorio “misc” y copiar dentro del mismo los siguientes archivos (utilice rutas absolutas con el comando *cp*):

```
/etc/passwd
/usr/share/sounds/alsa/Noise.wav
/bin/zcat
/bin/cpio
/usr/share/doc/zenity/NEWS.gz
/usr/share/doc//sudo/README
```

4. Una vez copiados, liste los archivos en el directorio *misc*. Usted debería observar que existen seis archivos:

```
$ pwd
$ ls -l
```

5. El comando “ls” anterior le ha entregado información sobre los archivos. Indique el tamaño de cada archivo (en bytes y en KiB).
6. Observe de qué tipo son los seis archivos en el directorio *misc*. Utilice el comando “file archivo”. Ejemplo:

```
$ file passwd
```

7. Ahora analizaremos el contenido de cada archivo. Para realizar esta tarea obtendremos los primeros 32 bytes de cada archivo. Utilizaremos el programa “hexdump” para visualizar en base 16 (hexadecimal) el valor de cada byte. Ejemplo:

```
$ hexdump -C cpio | head -2
```

El comando anterior le presenta en pantalla el contenido de los primeros 32 bytes del archivo *cpio*. Repita la operación para cada archivo en el directorio *misc*. Registre los resultados.

8. Escriba el contenido en base 2 (binario) de los primeros 4 bytes de cada archivo.
9. Sabiendo que cada archivo en el sistema es almacenado en una memoria que únicamente puede mantener los datos utilizando componentes electrónicos biestables (que mantiene únicamente dos valores diferentes) analice y desarrolle un texto que intente explicar cómo es posible que los archivos en el sistema sean de diferente “tipo” si la computadora mantiene únicamente los datos en base 2.

### 1.3. Anexo I: Conexión desde entorno Windows a usuario de alumno FI

1. Ingresamos en el buscador de Windows y ponemos la palabra *remoto*, lo que nos traerá como resultado la aplicación: “Conexión a Escritorio remoto”. Dicho procedimiento se ilustra en la Figura 2.

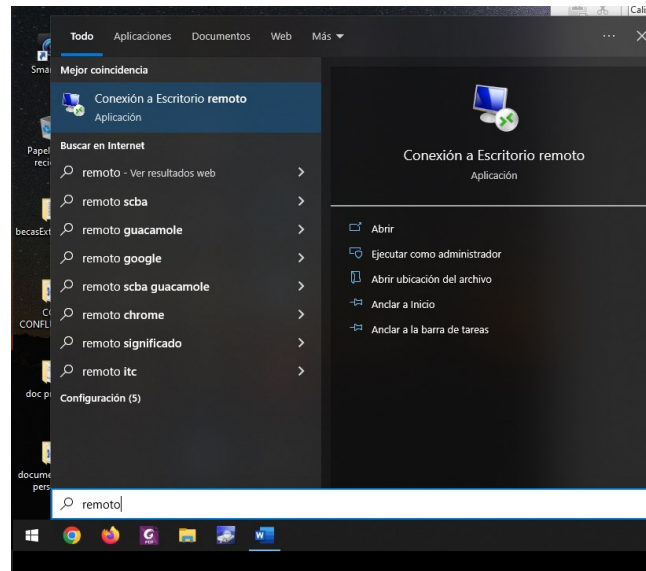


Figura 2: Búsqueda de Escritorio Remoto Windows.

2. Al hacer clic sobre el programa se nos abrirá la interfaz para ingresar la dirección a la cual queremos conectarnos según Figura 3.

En donde dice Equipo ingresamos: *aularemotafi.uncoma.edu.ar:1199*

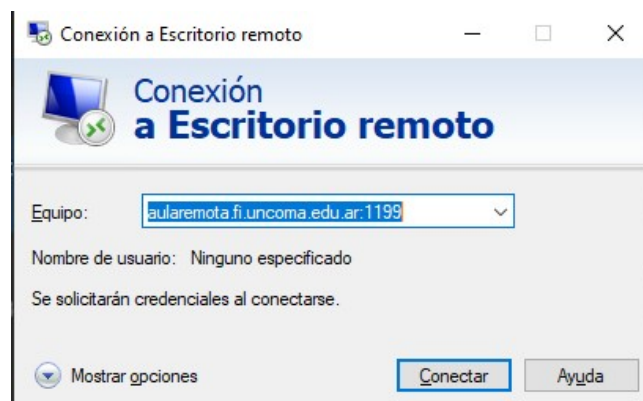


Figura 3: Pantalla de Escritorio Remoto Windows.

3. Luego nos aparecerá la pantalla de inicio de sesión del laboratorio Linux como se muestra en la Figura 4, donde tendrá que proceder con su usuario y contraseña oportunamente creado.
4. Una vez que esté logueado, para acceder a la Terminal, deberá ir a *Aplicaciones - Herramientas del sistema - Terminal* según Figura 5

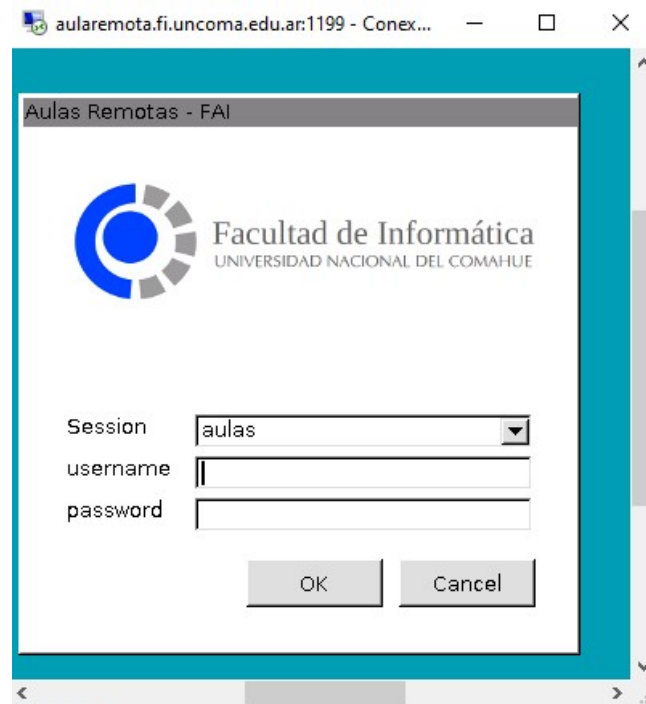


Figura 4: Logueo de Usuario Linux.

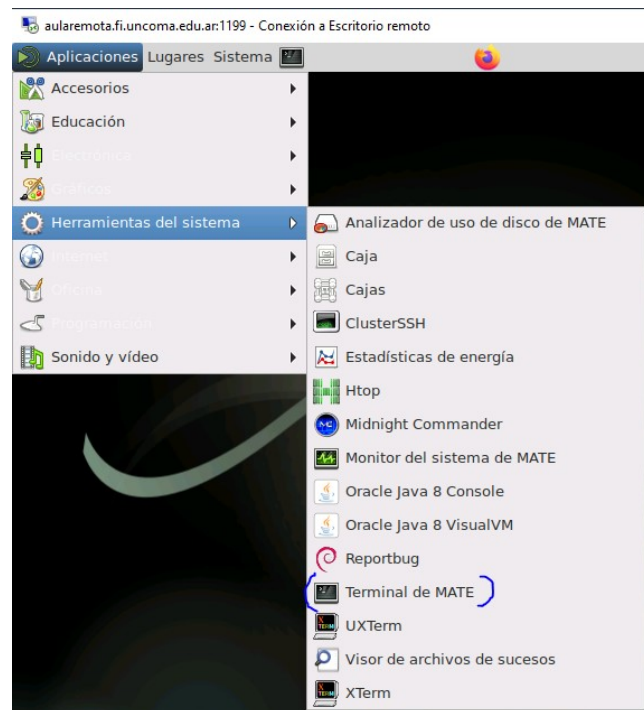


Figura 5: Búsqueda de terminal Linux.

## 1.4. Anexo II: Conexión desde entorno Linux a usuario de alumno FI

En GNU/Linux utilizaremos el comando `freerdp` para conectarnos al laboratorio de Introducción de Computación. Para instalar `freerdp`, desde una terminal ejecutar los siguientes comandos:

1. En Ubuntu: `sudo apt install freerdp2-x11 freerdp2-shadow-x11`
2. En Debian: `sudo apt install lightdm-remote-session-freerdp2`
3. Para conectarse a un escritorio remoto usando `xfreerdp`:  
`xfreerdp /u:USERNAME /p:PASSWORD /v:HOST[:PORT]`
4. Por ejemplo, para el usuario “mafalda” y su clave “12345”:  
`xfreerdp /u:mafalda /p:12345 +glyph-cache /v:aularemotafi.uncoma.edu:1199`
5. Una vez que esté logueado, para acceder a la Terminal, deberá ir a *Aplicaciones - Herramientas del sistema - Terminal* según Figura 6.

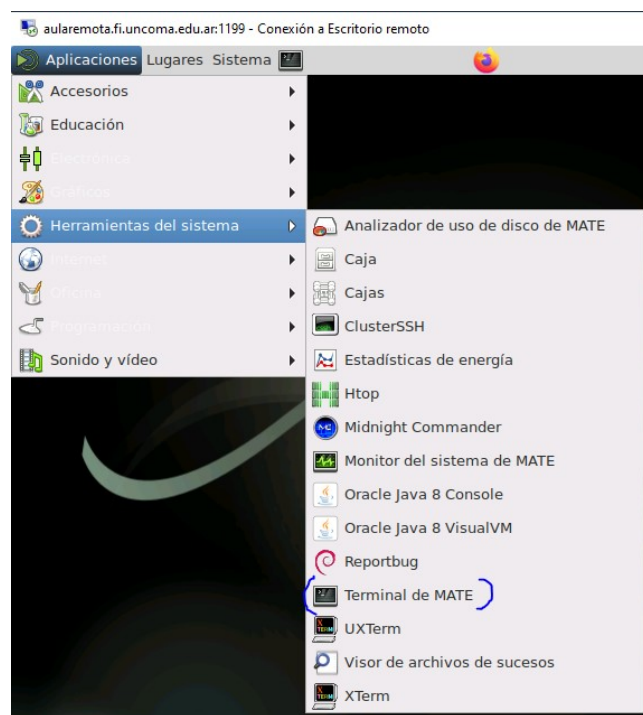


Figura 6: Búsqueda de terminal Linux.